SkillRack

MONICA A-1912080@nec

0/10 22 0 0 817 1495

Theme

Home Reports Profile Help Logout

LeaderBoard & Prev Day Solution

DAILY TEST ProgramID- 6377 SkillRack

Time Left: 00:17:34

Count of Concatenated Multiples

The program must accept an integer array of size **N** and an integer **X** as the input. The program must print the count of pairs in the array such that the concatenation(s) of two integers in the pair is a multiple of X. For example, the concatenations of 25 and 40 are 2540 and 4025.
Note: A pair cannot be formed with the integer at the same position in the array.

Boundary Condition(s):
2 <= N <= 100
1 <= X <= 999

Input Format:
The first line contains the value of N and X separated by space(s).
The second line contains N integers separated by space(s).

Output Format:
The first line contains the count of pairs which are the multiples of X.

Example Input/Output 1:
Input:
5 11
45 1 10 12 11

Output:
7

Explanation:
There are 7 pairs which produce the multiples of 11. They are
(45, 1) -> 451
(45, 10) -> 4510
(10, 45) -> 1045
(1, 10) -> 110
(12, 1) -> 121
(10, 12) -> 1012
(12, 10) -> 1210
Hence the output is 7

Example Input/Output 2:
Input:
3 12
12 12 11

Output:
2

Max Execution Time Limit: 5000 millisecs

Ambiance | Java (12.0) | Reset

```
1  import java.util.*;
2  public class Hello {
3
4      public static void main(String[] args) {
5              Scanner sc=new Scanner(System.in);
6              int n=sc.nextInt();
7              int m=sc.nextInt();
8              int[] arr=new int[n];
9              int c=0;
10             for(int i=0;i<n;i++){
11                 arr[i]=sc.nextInt();
12             }
13             for(int i=0;i<n;i++){
14                 for(int j=0;j<n;j++){
15                     if(i!=j){
16                 String a=arr[i]+""+arr[j];
17                 System.out.print(a+" ");
18                 if(Integer.parseInt(a)%m==0){
19                     c++;
```

```
                    }
                        }
                }
                }
                System.out.print(c);
        }
}
```

1912080@nec

Code did not pass the execution

Input:

```
4 2
12 78 42 10
```

Expected Output:

```
12
```

Your Program Output:

```
9
```

Save Run

Run with a custom test case (Input/Output)